W/53

esp@cenet

STORAGE SUB-SYSTEM AND ITS CONTROL METHOD

Patent Number: JP2001067187
Publication date: 2001-03-16
Inventor(s): ARAKAWA TAKASHI; MOGI KAZUHIKO; YAMAKAMI KENJI; ARAI HIROHARU
Applicant(s):: HITACHI LTD
Requested Patent: JP2001067187 (JP01067187)
Application Number: JP19990242713 19990830
Priority Number(s):
IPC Classification: G06F3/06 ; G06F12/00
EC Classification:
Equivalents:

Abstract

PROBLEM TO BE SOLVED: To simplify a work for optimizing arrangement by re-arrangement by the user of a disk array system or the like by changing the correspondence of a logical storage area from a physical storage area into the second physical storage area and executing re-arrangement.
SOLUTION: A control part 300 automatically executes re-arrangement execution processing at the set time and date. That is, the part 300 copies contents stored in a re-arrangement source physical area in a re-arrangement destination physical area based on re-arrangement information 408. Moreover, at the point of time when the copying is completed and the whole contents of the re-arrangement source physical area are reflected in the re-arrangement destination physical area, the control part 300 changes a physical area corresponding to a logical area for executing re-arrangement in logical/physical correspondence information 400 from the re-arrangement source physical area into the re-arrangement destination physical area. Besides, the control part 300 uses the re-arrangement destination physical area on a non-usage physical area 1470, changes the re-arrangement source physical area into the non-usage one and, moreover, updates the time and date of re-arrangement execution time information 406 into the one for a next time by referring to time and date updating information on re-arrangement execution time information 406.

(19)日本国特許庁（ＪＰ）　(12)公開特許公報（Ａ）　(11)特許出願公開番号
特開2001－67187
（P2001－67187A）
(43)公開日　平成13年３月16日(2001.3.16)

(51)Int.Cl.'	識別記号	ＦＩ		テーマコード'（参考）
Ｇ０６Ｆ　3/06	３０１	Ｇ０６Ｆ　3/06	３０１Ａ	５Ｂ０６５
	５４０		５４０	５Ｂ０８２
12/00	５０１	12/00	５０１Ｂ	

審査請求　未請求　請求項の数10　ＯＬ　（全 24 頁）

(21)出願番号　特願平11－242713

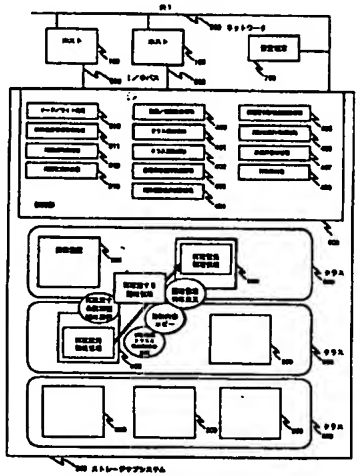
(22)出願日　平成11年８月30日(1999.8.30)

(71)出願人　000005108
株式会社日立製作所
東京都千代田区神田駿河台四丁目６番地
(72)発明者　荒川　敬史
神奈川県川崎市麻生区王禅寺1099番地　株式会社日立製作所システム開発研究所内
(72)発明者　茂木　和彦
神奈川県川崎市麻生区王禅寺1099番地　株式会社日立製作所システム開発研究所内
(74)代理人　100075096
弁理士　作田　康夫

最終頁に続く

(54)【発明の名称】　ストレージサブシステム及びその制御方法

(57)【要約】
【課題】ストレージサブシステムのユーザまたは保守員が記憶領域の物理的再配置による配置最適化を行うための作業を簡便にするストレージサブシステムおよび制御方法を提供する。
【解決手段】ストレージサブシステム２００は、記憶装置５００を、それぞれ属性を有する複数の組（クラス）６００として管理し、クラス属性に基づき好適な再配置先のクラスを決定する。



【特許請求の範囲】
【請求項１】複数の記憶装置と、前記記憶装置の使用状況情報を取得する手段と、前記計算機がリードライト対象とする論理記憶領域と前記記憶装置の第一の物理記憶領域との対応づけを行う手段とを有し、１台以上の計算機に接続するストレージサブシステムの制御方法であって、前記記憶装置は複数の組（クラス）に分類され、前記クラスは設定された属性を有し、前記ストレージサブシステムは、前記使用状況情報および前記クラス属性に基づき前記論理記憶領域に好適な再配置先のクラスを決定し、前記論理記憶領域の再配置先として利用可能な第二の物理記憶領域を前記クラス内から選択し、前記第一の物理記憶領域の内容を前記第二の前記物理記憶領域にコピーするとともに論理記憶領域の対応づけを前記第一の物理記憶領域から前記第二の物理記憶領域へ変更して再配置を行うことを特徴とする前記ストレージサブシステムの制御方法。
【請求項２】請求項１に記載のストレージサブシステムの制御方法であって、ストレージサブシステムは、前記使用状況情報を蓄積し、設定された期間の前記使用状況情報に基づき、論理記憶領域の再配置先を決定し、設定された時間に再配置を行うことを特徴とするストレージサブシステムの制御方法。
【請求項３】請求項１または２に記載のストレージサブシステムの制御方法であって、ストレージサブシステムは、使用状況情報として、記憶装置の単位時間当たりの使用時間（使用率）を用い、各クラスは、属性として設定されたクラス間の性能順位と使用率上限値を有し、前記ストレージサブシステムは、クラスの使用率上限値を超えている記憶装置から再配置する論理記憶領域を選択し、前記論理記憶領域の再配置先のクラスを前記順位の上位のクラスから、各クラスの使用率上限値を超えないように決定することを特徴とするストレージサブシステムの制御方法。
【請求項４】請求項１または２に記載のストレージサブシステムの制御方法であって、ストレージサブシステムは、使用状況情報として、記憶装置の単位時間当たりの使用時間（使用率）を用い、各クラスは、属性として設定されたクラス間の性能順位と使用率上限値を有し、前記ストレージサブシステムは、クラスの使用率上限値を超えている記憶装置から再配置する論理記憶領域を選択し、前記論理記憶領域の再配置先として利用可能な物理記憶領域を同一クラス内の記憶装置から、前記クラスの使用率上限値を超えないように決定することを特徴とするストレージサブシステムの制御方法。
【請求項５】請求項１または２に記載のストレージサブシステムの制御方法であって、ストレージサブシステムは、使用状況情報として、記憶装置の単位時間当たりの使用時間（使用率）を用い、各クラスは属性として設定された対象アクセス種別と使用率上限値を有し、前記ストレージサブシステムは、クラスの使用率上限値を超えている記憶装置から再配置する論理記憶領域を選択し、前記論理記憶領域に対するアクセス種別の分析結果に基づいて前記論理記憶領域の再配置先のクラスを前記対象アクセス種別のクラスから、各クラスの使用率上限値を超えないように決定することを特徴とするストレージサブシステムの制御方法。
【請求項６】１台以上の計算機に接続し、複数の記憶装置と、前記記憶装置の使用状況情報を取得する手段と、前記計算機がリードライト対象とする論理記憶領域と前記記憶装置の第一の物理記憶領域との対応づけを行う手段とを有するストレージサブシステムであって、前記複数のディスク装置をそれぞれ属性を有する複数の組（クラス）として管理する手段と、前記使用状況情報および前記クラス属性に基づき前記論理記憶領域に好適な再配置先のクラスを決定する手段と、前記論理記憶領域の再配置先として利用可能な第二の物理記憶領域を前記クラス内から選択する手段と、前記第一の物理記憶領域の内容を前記第二の前記物理記憶領域にコピーするとともに論理記憶領域の対応づけを前記第一の物理記憶領域から前記第二の物理記憶領域へ変更して再配置を行う手段とを有することを特徴とするストレージサブシステム。
【請求項７】請求項６に記載のストレージサブシステムであって、ストレージサブシステムは、前記使用状況情報を蓄積し、設定された期間の前記使用状況情報に基づき、論理記憶領域の再配置先を自動的に決定する手段と、設定された時間に再配置を行う手段とを有することを特徴とするストレージサブシステム。
【請求項８】請求項６または７に記載のストレージサブシステムであって、ストレージサブシステムは、使用状況情報として記憶装置の単位時間当たりの使用時間（使用率）を用いる手段を有し、前記ストレージサブシステムは、各クラスに属性として設定されている使用率上限値を超えている記憶装置から再配置する論理記憶領域を選択する手段と、前記論理記憶領域の再配置先のクラスを各クラスに属性として設定されているクラス間の性能順位から、各クラスの使用率上限値を超えないように決定する手段とを有することを特徴とするストレージサブシステム。
【請求項９】請求項６または７に記載のストレージサブシステムであって、ストレージサブシステムは、使用状況情報として、記憶装置の単位時間当たりの使用時間（使用率）を用いる手段を有し、前記ストレージサブシステムは、属性として設定されたクラスの使用率上限値を超えている記憶装置から再配置する論理記憶領域を選択する手段と、前記論理記憶領域に対するアクセス種別を分析する手段と、対象アクセス種別を属性として設定されたクラスから、前記論理記憶領域の再配置先のクラスを前記分析結果に基づいて各クラスの使用率上限値を超えないように決定する手段とを有することを特徴とす

1453

るストレージサブシステム。

【請求項10】請求項6、7、8、または9に記載のストレージサブシステムであって、ストレージサブシステムは、複数のディスク装置を有するディスクアレイであり、前記ディスク装置の使用率を使用状況情報として用いる手段を有することを特徴とするストレージサブシステム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、複数の記憶装置を有するストレージサブシステム、およびその制御方法に関する。

【0002】

【従来の技術】コンピュータシステムにおいて、高性能を実現する二次記憶システムの1つにディスクアレイシステムがある。ディスクアレイシステムは、複数のディスク装置をアレイ状に配置し、前記各ディスク装置に分割格納されるデータのリード/ライトを、前記各ディスク装置を並列に動作させることによって、高速に行うシステムである。ディスクアレイシステムに関する論文としては、D. A. Patterson, G. Gibson, and R. H. Katz, "A Case for Redundant Arrays of Inexpensive Disks (RAID)" (in Proc. ACM SIGMOD, pp. 109-116, June 1988)がある。この論文では、冗長性を付加したディスクアレイシステムに対し、その構成に応じてレベル1からレベル5の種別を与えている。これらの種別に加えて、冗長性無しのディスクアレイシステムをレベル0と呼ぶこともある。上記の各レベルは冗長性などにより実現するためのコストや性能特性などが異なるため、ディスクアレイシステムを構築するにあたって、複数のレベルのアレイ（ディスク装置の組）を混在させることも多い。ここでは、この組のことをパリティグループと呼ぶ。

【0003】ディスク装置は、性能や容量などによりコストが異なり、ディスクアレイシステムを構築するにあたって最適なコストパフォーマンスを実現するために、やはり性能や容量の異なる複数種のディスク装置を用いることがある。

【0004】ディスクアレイシステムに格納されるデータを上記のようにディスク装置に分散して配置するため、ディスクアレイシステムは、ディスクアレイシステムに接続するホストコンピュータがアクセスする論理記憶領域とディスク装置の記憶領域を示す物理記憶領域の対応づけ（アドレス変換）を行う。特開平9-274544号公報には、ホストコンピュータからの論理記憶領域に対するI/Oアクセスについての情報を取得する手段と、論理記憶領域の物理記憶領域への対応づけを変更して物理的再配置を行う手段により、格納されたデータの最適配置を実現するディスクアレイシステムが開示されている。

【0005】

【発明が解決しようとする課題】特開平9-274544号公報に示されるような従来の技術における配置最適化の実行方法については以下の課題がある。

【0006】再配置する論理記憶領域の選択および再配置先の物理記憶領域の選択にあたり、ディスクアレイシステムのユーザまたは保守員が、前記ディスクアレイシステムの構成や個々のディスク装置の特性や性能などの情報を確認して前記選択を行わなければならず、ユーザまたは保守員による作業が煩雑となっていた。

【0007】また、ディスクアレイシステムが選択を自動的に行う場合においても、ユーザまたは保守員が前記個々のディスク装置の情報を確認して選択基準値を規定しなければならず、やはりユーザまたは保守員による作業が煩雑となっていた。特に、上記のように異種のレベルや異種のディスク装置の混在するディスクアレイシステムについては情報管理の煩雑さが増大する。

【0008】また、ディスクアレイシステムが選択のために行うI/Oアクセス情報の参照は、ホストコンピュータおよびディスクアレイシステムを含むシステムで行われる処理のスケジュールの特性を考慮していなかった。一般にコンピュータシステムで行われる処理と処理に伴うI/Oは、ユーザによって作成されたスケジュールに則って行われており、また処理およびI/Oの傾向は日毎、月毎、年毎などの周期性を示す場合も多く、一般にユーザは特定期間の処理およびI/Oに関心があると考えられる。

【0009】また上記従来技術において、再配置による性能チューニング方法については以下の課題がある。物理的再配置による性能チューニング方法は、ディスク装置、すなわち、物理記憶領域の使用状況に変更を加えるものであるが、従来の技術においては、ホストコンピュータからの論理記憶領域に対するI/Oアクセスについての情報を参照するため、再配置する論理記憶領域の選択および再配置先の物理記憶領域の選択にあたり、正しい選択が行えない可能性があった。

【0010】また、ホストコンピュータからのシーケンシャルアクセスとランダムアクセスが顕著に、同一のディスク装置に含まれる別々の物理記憶領域に対して行われる場合でも、シーケンシャルアクセスとランダムアクセスを異なるディスク装置に分離するために、再配置先のディスク装置を任意に特定して自動的再配置を行わせることはできなかった。一般に、ホストコンピュータからの処理要件として、データ長の小さいランダムアクセスには短時間での応答（高応答性能）が求められるが、同一ディスク装置にデータ長の大きいシーケンシャルアクセスが存在する場合、ランダムアクセスの応答時間はシーケンシャルアクセスの処理に阻害されて長くなり、

応答性能は悪化してしまう。

【0011】本発明の第一の目的は、ディスクアレイシステムのユーザまたは保守員が再配置による配置最適化を行うための作業を簡便にすることにある。

【0012】本発明の第二の目的は、ホストコンピュータおよびディスクアレイシステムを含むシステムでの処理のスケジュールを考慮した再配置による配置最適化を可能にすることにある。

【0013】本発明の第三の目的は、再配置する論理記憶領域の選択および再配置先の物理記憶領域の選択にあたり、実際の記憶装置であるディスク装置の使用状況に基づく選択を行う、ディスクアレイシステムの制御方法およびディスクアレイシステムを提供することにある。

【0014】本発明の第四の目的は、ディスクアレイシステムにおける同一ディスク装置での顕著なシーケンシャルアクセスとランダムアクセスの混在に対し、再配置先のディスク装置を任意に特定して再配置によりシーケンシャルアクセスおよびランダムアクセスを異なるディスク装置に自動的に分離することができるようにすることにある。

【0015】

【課題を解決するための手段】上記の第一目的を実現するために、1台以上のホストコンピュータに接続するディスクアレイシステムは、配下の複数のディスク装置の使用状況情報を取得する手段と、ホストコンピュータがリード/ライト対象とする論理記憶領域とディスク装置の第一の物理記憶領域との対応づけを行う手段とを有し、さらに、複数のディスク装置をそれぞれ属性を有する複数の組（クラス）として管理する手段と、使用状況情報およびクラス属性に基づき論理記憶領域に好適な再配置先のクラスを決定する手段と、論理記憶領域の再配置先として利用可能な第二の物理記憶領域をクラス内から選択する手段と、第一の物理記憶領域の内容を前記第二の前記物理記憶領域にコピーするとともに論理記憶領域の対応づけを第一の物理記憶領域から第二の物理記憶領域へ変更して再配置を行う手段を備える。

【0016】また、上記第二の目的を実現するために、ディスクアレイシステムは、使用状況情報を蓄積し、設定された期間の使用状況情報に基づき、論理記憶領域の再配置先を決定する手段と、設定された時間に再配置を行う手段を備えることができる。

【0017】また、上記第三の目的を実現するために、ディスクアレイシステムは、使用状況情報として、ディスク装置の単位時間当たりの使用時間（使用率）を用いる手段を備える。

【0018】また、上記第四の目的を実現するために、ディスクアレイシステムは、各クラスに属性として設定された対象アクセス種別（シーケンシャル/ランダムアクセス種別）と使用率上限値を用いて、クラスの使用率上限値を超えている記憶装置から再配置する論理記憶領域を選択し、論理記憶領域に対するアクセス種別の分析結果に基づいて論理記憶領域の再配置先のクラスを好適なアクセス種別のクラスから、各クラスの使用率上限値を超えないように決定する手段を備える。

【0019】

【発明の実施の形態】以下、本発明の実施の形態を図1～図27を用いて説明する。

【0020】<第一の実施の形態>本実施の形態では、クラス600に基づく再配置の判断と、再配置判断および実行のスケジューリングについて説明する。

【0021】図1は、本発明の第1の実施の形態における計算機システムの構成図である。

【0022】本実施の形態における計算機システムは、ホスト100、ストレージサブシステム200、制御端末700を有してなる。

【0023】ホスト100は、ストレージサブシステム200にI/Oパス800を介して接続し、ストレージサブシステム200に対しリードやライトのI/Oを行う。I/Oの際、ホスト100は、ストレージサブシステム200の記憶領域について論理領域を指定する。I/Oパス800の例としては、ESCON、SCSI、ファイバチャネルなどがある。

【0024】ストレージサブシステム200は、制御部300および複数の記憶装置500を有する。制御部300は、リード/ライト処理310、使用状況情報取得処理311、再配置判断処理312、及び再配置実行処理313を行う。また、ストレージサブシステム200は、論理/物理対応情報400、クラス構成情報401、クラス属性情報402、論理領域使用状況情報403、物理領域使用状況情報404、再配置判断対象期間情報405、再配置実行時刻情報406、未使用領域情報407、及び再配置情報408を保持する。

【0025】ホスト100、制御部300、および制御端末700は、ネットワーク900で接続される。ネットワーク900の例としては、FDDI、ファイバチャネルなどがある。

【0026】ホスト100、制御部300、および制御端末700には、各々での処理を行うためのメモリ、CPUなど、計算機において一般に用いられる構成要素もそれぞれ存在するが、本実施の形態の説明においては重要でないため、ここでは説明を省略する。

【0027】ホスト100が、ストレージサブシステム200に対してリード/ライトを行う場合のリード/ライト処理310、および使用状況情報取得処理311について図2で説明する。

【0028】リード/ライト処理310において、ホスト100は、ストレージサブシステム200の制御部300に対しリードまたはライトを論理領域を指定して要求する（ステップ1000）。要求を受領した制御部300は、論理/物理対応情報400を用いて論理領域に

対応する物理領域を求め、すなわち論理領域のアドレス（論理アドレス）を物理領域のアドレス（物理アドレス）に変換する（ステップ1010）。続いて制御部300は、リードの場合は、この物理アドレスの記憶装置500からデータを読み出してホスト100に転送し、ライトの場合は、ホスト100から転送されたデータを前記物理アドレスの記憶装置500に格納し（ステップ1020）、さらに後述の使用状況情報取得処理311を行う。リード/ライト要求およびデータ転送はI/Oバス800を介して行われる。

【0029】論理/物理対応情報400の一例を図3に示す。論理アドレスはホスト100がリード/ライト処理310で用いる論理領域を示すアドレスである。物理アドレスは実際にデータが格納される記憶装置500上の領域を示すアドレスであり、記憶装置番号および記憶装置内アドレスからなる。記憶装置番号は個々の記憶装置500を示す。記憶装置内アドレスは記憶装置500内での記憶領域を示すアドレスである。

【0030】次に、使用状況情報取得処理311において制御部300は、リード/ライト処理310においてリード/ライト対象となった論理領域についての論理領域使用状況情報403と、リード/ライト処理310で使用した物理領域についての物理領域使用状況情報404を更新する（ステップ1030、1040）。論理領域使用状況情報403および物理領域使用状況情報404は、例えば使用頻度、使用率、リード/ライトに関する属性など、各々の論理領域と物理領域の各日時の使用状況に関する情報である。論理領域使用状況情報403および物理領域使用状況情報404の具体的な例は、以降の実施の形態で説明する。

【0031】次に、制御部300が行う再配置判断処理312について図4で説明する。

【0032】記憶装置500は、ユーザによって、または初期状態として複数の組（クラス600）に分類されており、クラス600への分類はクラス構成情報401に設定されている。さらに、各クラス600は、ユーザによって、または初期条件として属性を設定されており、属性は、クラス属性情報402に設定されている。クラス属性情報402は、許容使用状況や好適な使用状況やクラス間優先順位などの属性に関する情報である。クラス構成情報401およびクラス属性情報402の具体的な例は、以降の実施の形態で説明する。再配置判断対象期間情報405には、ユーザによってまたは初期条件として再配置判断処理312の対象とする使用状況情報の期間と期間更新情報が設定されている。

【0033】再配置判断対象期間情報405の一例を図5に示す。開始日時から終了日時までの期間が対象期間となる。期間更新情報は次回の対象期間の設定条件であり、例えば毎週、毎日、X時間後などがありうる。制御部300は、対象期間の論理領域使用状況情報403および物理領域使用状況情報404を参照し（ステップ1100）、クラス属性情報402の各クラス600の許容使用状況などと比較して（ステップ1110）、物理的再配置を行うべき論理領域を選択する（ステップ1120）。

【0034】さらに、制御部300は、クラス属性情報402の許容使用状況や好適な使用状況やクラス間優先順位などを参照して（ステップ1130）、論理領域の再配置先のクラス600を選択し（ステップ1140）、さらに、クラス600に属する記憶装置500の中から論理領域の再配置先として未使用の物理領域を選択し（ステップ1150）、選択結果を再配置情報408に出力する（ステップ1160）。

【0035】再配置情報408の一例を図6に示す。論理領域は、再配置する論理領域であり、再配置元物理領域は、論理領域に対応する現在の物理領域を示す記憶装置番号と記憶装置内アドレスであり、再配置先物理領域は、再配置先の物理領域を示す記憶装置番号と記憶装置内アドレスである。図6に示すように再配置の立案は一つ以上行われうる。さらに制御部300は、再配置判断対象期間情報405の期間更新情報を参照して、再配置判断対象期間情報405の対象期間を次回分に更新する（ステップ1170）。上記の処理において制御部300は、論理/物理対応情報400を用い、また前記の未使用の物理領域の検索に未使用領域情報407を用いる。

【0036】未使用領域情報407の一例を図7に示す。記憶装置番号は個々の記憶装置500を示す。記憶装置内アドレスは記憶装置500内での領域を示すアドレスである。記憶装置番号および装置内アドレスは物理領域を示し、使用/未使用の項目は、物理領域の使用/未使用の区別を示す。制御部300は、通常、再配置判断処理312を対象期間以後、後述の再配置実行処理313以前に自動的に行う。

【0037】次に、制御部300が行う再配置実行処理313について図8で説明する。

【0038】再配置実行時刻情報406にはユーザによってまたは初期条件として再配置実行処理313を行う日時と日時更新情報が設定されている。

【0039】再配置実行時刻情報406の一例を図9に示す。制御部309は、設定された日時に以下に説明する再配置実行処理313を自動的に実行する。日時更新情報は次回の再配置実行処理313を行う日時の設定条件であり、例えば毎週、毎日、X時間後などがありうる。制御部300は、再配置情報408に基づき再配置元物理領域に格納している内容を再配置先物理領域にコピーする（ステップ1200）。さらに、コピーが完了して再配置元物理領域の内容が全て再配置先物理領域に反映された時点で、制御部300は、論理/物理対応情報400上の再配置を行う論理領域に対応する物理領域を再配置元物理領域から再配置先物理領域に変更する（ステップ1210）。

【0040】さらに、制御部300は、未使用物理領域470上の再配置先物理領域を使用とし、再配置元物理領域を未使用に変更する（ステップ1220）。さらに制御部300は、再配置実行時刻情報406の日時更新情報を参照して、再配置実行時刻情報406の日時を次回分に更新する（ステップ1230）。

【0041】ユーザまたは保守員は、制御部300が上記の処理で用いている各情報を、制御端末700からネットワーク900を介して、またはホスト100からネットワーク900またはI/Oバス800を介して設定および確認すること、特に、再配置情報408を確認および設定して再配置案を修正や追加や削除などをすることができる。

【0042】上記の処理を行うことによって、取得した使用状況情報および設定されたクラス属性に基づいて、ストレージサブシステム200において論理領域の物理的再配置を自動的に行い、ストレージサブシステム200の最適化を行うことができる。さらに上記の再配置判断および実行の処理を繰り返して配置を修正していくことによって、使用状況の変動やその他の最適化誤差要因を吸収していくことができる。

【0043】特に、上記の処理により、ユーザまたは保守員は再配置による最適化を簡便に行うことができる。ユーザまたは保守員は、記憶装置500をクラス600という単位で管理できるため、記憶装置500の性能や信頼性や特性などの属性を個々の前記記憶装置500について管理する必要はない。さらに、ユーザまたは保守員は、記憶装置500の個々の属性が等しくない組に対しても、必要に応じて同一の属性を持つクラス600を設定して、1つの管理単位として扱うことができる。ただし、1つの記憶装置500が1つのクラス600を構成すると見なして1つの記憶装置500を管理単位として上記の再配置の処理を行うことも可能である。

【0044】また、ユーザまたは保守員は、ホスト100で行われる処理（ジョブ）の特徴やスケジュールを考慮して、上記の再配置を自動的に行うことができる。一般に、計算機システムで行われる処理と、この処理に伴うI/Oは、ユーザによって作成されたスケジュールに則って行われる。ユーザは、特に最適化の対象としたい処理を有する場合、処理の期間を特定することが可能であり、本実施の形態で説明した再配置の処理によって、ユーザは関心のある期間を指定して再配置判断の処理をストレージシステム200に行わせ、すなわち、前記期間の使用状況情報に基づいて上記の再配置による最適化を実現することができる。また、計算機システムで行われる処理およびI/Oの傾向は日毎、月毎、年毎などの周期性を示す場合も多い。特に、処理が定型業務に基づく処理である場合には、周期性が顕著となる。前述の場合と同様にユーザは、周期において特に最適化対象として関心のある期間を指定して再配置による最適化を行うことができる。また、再配置実行処理313では、ストレージシステム200内で格納内容のコピーを伴うが、ユーザはストレージシステム200があまり使用されていない時刻やホスト100で実行されている処理の要求処理性能が低い期間を再配置実行処理313の実行時刻として設定することで、ホスト100での要求処理性能が高い処理のストレージシステム200へのI/Oがコピーにより阻害されることを回避できる。

【0045】なお、記憶装置500は、それぞれ異なる性能、信頼性、特性や属性を持っていてよく、特に具体的には、磁気ディスク装置、磁気テープ装置、半導体メモリ（キャッシュ）のように異なる記憶媒体であってもよい。また、上記の例では未使用領域情報407は物理領域に基づいて記述されているとしたが、未使用の物理領域に対応する論理領域（論理アドレス）に基づいて記述されていてもよい。

【0046】<第二の実施の形態>本実施の形態では、使用状況情報としてのディスク装置使用率の適用と、クラス600の上限値およびクラス600間の性能順位による再配置判断について説明する。

【0047】図10は、本発明の第2の実施の形態における計算機システムの構成図である。

【0048】本実施の形態の計算機システムは、ホスト100、ディスクアレイシステム201、制御端末700を有してなる。本実施の形態における計算機システムは、第1の実施の形態でのストレージサブシステム200をディスクアレイシステム201とし、記憶装置500をパリティグループ501としたものに相当する。

【0049】ディスクアレイシステム201は、制御部300とディスク装置502を有する。制御部300は、第1の実施の形態での制御部300に相当する。ディスク装置502は、n台（nは2以上の整数）でRAID（ディスクアレイ）を構成しており、このn台のディスク装置502による組をパリティグループ501と呼ぶ。RAIDの性質として、1つのパリティグループ501に含まれるn台のディスク装置502は、n-1台のディスク装置502の格納内容から生成される冗長データが残りの1台に格納されるといった冗長性上の関係を持つ。またn台のディスク装置502は、冗長データを含めた格納内容が、並列動作性向上のためにn台のディスク装置502に分散格納されるなど、データ格納上の関係を持つ。この関係から各パリティグループ501を動作上の1単位とみなすことができるが、冗長性や台数nなどにより実現するためのコストや性能特性などが異なるため、ディスクアレイシステム201を構成するにあたって、レベルや台数nの異なるアレイ（パリティグループ501）を混在させることも多く、またパリティグループ501を構成するディスク装置502につ

いても、性能や容量などによりコストが異なるため、ディスクアレイシステム201を構成するにあたって最適なコストパフォーマンスを実現するために性能や容量の異なる複数種のディスク装置502を用いることもある。よって本実施の形態においてディスクアレイシステム201を構築する各パリティグループ501は性能、信頼性、特性などの属性が同一であるとは限らず、特に性能について差異があるとする。

【0050】本実施の形態における論理／物理対応情報400の一例を図11に示す。

【0051】論理アドレスは、ホスト100がリード／ライト処理310で用いる論理領域を示すアドレスである。物理アドレスは実際にデータと前記冗長データが格納されるディスク装置502上の領域を示すアドレスであり、パリティグループ番号と各々のディスク装置番号およびディスク装置内アドレスからなる。パリティグループ番号は個々のパリティグループ501を示す。ディスク装置番号は個々のディスク装置502を示す。ディスク装置内アドレスはディスク装置502内での領域を示すアドレスである。制御部300は、RAIDの動作として、冗長データに関する情報を前記リード／ライト処理310などで用いて処理するが、本実施の形態の説明では、パリティグループ502を動作上の1単位として説明するため、前記処理に関してはここでは特にふれない。

【0052】さらに第1の実施の形態と同様に、パリティグループ501は、ユーザによってまたは初期状態として複数の組（クラス600）に分類されており、クラス600への分類はクラス構成情報401に設定されている。クラス構成情報401の一例を図12に示す。

【0053】クラス番号は各クラス600を示す番号である。パリティグループ数は各クラス600に属するパリティグループの数を示す。パリティグループ番号は各クラス600に属するパリティグループ番号501を示す。同様に、各クラス600の属性は、クラス属性情報402に設定されている。本実施の形態におけるクラス属性情報402の一例を図13に示す。

【0054】クラス番号は、各クラス600を示す番号である。使用率上限値は後述のディスク使用率の許容範囲を示す上限値であり、クラス600の属するパリティグループ501に適用する。クラス間性能順位は、クラス600間の性能順位（数字の小さいものが高性能とする）である。クラス間性能順位は各クラス600を構成するパリティグループ501の前述の性能差異に基づく。再配置実行上限値および固定については後述する。

【0055】本実施の形態における使用状況情報取得処理311について図14で説明する。

【0056】制御部300は、第1の実施の形態と同様に、リード／ライト処理310において使用したディスク装置502の使用時間を取得して単位時間当たりの使用時間（使用率）を求め、さらに、ディスク装置502が属するパリティグループ501について、使用率の平均を算出し（ステップ1300）、使用率平均を、リード／ライト対象となった論理領域についてのディスク装置使用率として論理領域使用状況情報403に記録する（ステップ1310）。また制御部300は、パリティグループ501に対応する全論理領域のディスク装置使用率の和を求め（ステップ1320）、パリティグループ501の使用率として物理領域使用状況情報404に記録する（ステップ1330）。

【0057】本実施の形態における論理領域使用状況情報403および物理領域使用状況情報404の一例を図15および図16に示す。

【0058】日時はサンプリング間隔（一定期間）毎の日時を示し、論理アドレスは論理領域を示し、パリティグループ番号は個々のパリティグループを示し、論理領域のディスク装置使用率およびパリティグループ使用率はそれぞれ前記サンプリング間隔での平均使用率を示す。上記のようなディスク装置502の使用率はディスク装置502にかかる負荷を示す値であり、使用率が大きい場合は、ディスク装置502が性能ボトルネックとなっている可能性があるため、再配置処理で使用率を下げることによりディスクアレイシステム201の性能向上が期待できる。

【0059】次に、再配置判断処理312について図17で説明する。

【0060】制御部300は、各クラス600について、クラス600に属するパリティグループ501をクラス構成情報401から取得する（ステップ1300）。続いて、制御部300は、第1の実施の形態と同様の再配置判断対象期間情報405を参照して対象期間を取得し、さらにパリティグループ501について、対象期間の物理領域使用状況情報404のパリティグループ使用率を取得し集計する（ステップ1320）。続いて、制御部300は、クラス属性情報402を参照してクラス600の使用率上限値を取得する（ステップ1330）。制御部300は、パリティグループ使用率とクラス上限値を比較し、パリティグループ使用率がクラス上限値より大きい場合は、パリティグループ501の使用率を減らすために、パリティグループ501に対応する論理領域の再配置が必要と判断する（ステップ1340）。

【0061】続いて、制御部300は、対象期間の論理領域使用状況情報403を参照して、再配置が必要と判断したパリティグループ501の各物理領域に対応する論理領域のディスク装置使用率を取得し集計して（ステップ1350）、ディスク装置使用率の大きいものから、再配置する論理領域として選択する（ステップ1360）。論理領域の選択は、パリティグループ501の使用率から選択した論理領域のディスク使用率を減算していき、クラス600の使用率上限値以下になるまで行う（1370）。ディスク装置使用率の大きい論理領域は、パリティグループ501の使用率に対する影響も大きく、またホスト100からの論理領域に対するアクセス頻度も大きいと考えられるため、ディスク装置使用率の大きい論理領域を優先的に再配置することで、ディスクアレイシステム201の効果的な性能改善が期待できる。

【0062】制御部300は、選択された論理領域についての再配置先となる物理領域を探す。制御部300は、クラス属性情報402を参照し、パリティグループ501の属するクラス600より性能順位が高位のクラス600（高性能クラス）に注目し、クラス構成情報401および第1の実施の形態と同様の未使用領域情報407を参照して高性能クラスに属するパリティグループ501の未使用物理領域を取得する（ステップ1380）。

【0063】さらに、制御部300は、各未使用物理領域について、再配置先とした場合のパリティグループ使用率の予測値を求め（ステップ1390）、未使用物理領域の中から、再配置先とした場合に高性能クラスに設定されている上限値を超えないと予測できる未使用物理領域を、再配置先の物理領域として選択し（ステップ1400）、選択結果を第1の実施の形態と同様に、再配置情報408に出力する（ステップ1410）。選択した全ての論理領域について再配置先の物理領域を選択し終えたら処理を終了する（ステップ1420）。

【0064】本実施の形態において、制御部300は、第1の実施の形態に加えてパリティグループ情報409を保持し、パリティグループ情報409、論理領域使用状況情報403、及び物理領域使用状況情報404から使用率予測値を算出する。

【0065】パリティグループ情報409の一例を図18に示す。パリティグループ番号は個々のパリティグループ501を示す番号である。RAID構成はパリティグループ501が構成するRAIDのレベルやディスク台数や冗長度構成を示す。ディスク装置性能はパリティグループ501を構成するディスク装置502の性能特性を示す。固定については後述する。上記の処理においてディスク装置使用率の大きい論理領域の再配置先を高性能クラスのパリティグループ501とすることで、同一負荷に対するディスク装置使用時間を短縮でき、論理領域の再配置後のディスク装置使用率を抑制できる。

【0066】再配置実行処理313は、第1の実施の形態と同様に行われるが、図19に示すように、制御部300は、再配置のためのコピーを行う前にクラス属性情報402を参照し、再配置元および再配置先のクラス600について、ユーザによってまたは初期条件として設定された再配置実行上限値を取得する（ステップ1500）。さらに物理領域使用状況情報404を参照して、再配置元および再配置先のパリティグループ501の直近のパリティグループ使用率を取得し（ステップ1510）、比較の結果少なくとも一方のクラス600においてパリティグループ使用率が再配置実行上限値を超えていた場合は（ステップ1520、1530）、再配置実行処理313を中止または延期する（ステップ1540）。

【0067】上記処理によりユーザは、パリティグループ501の使用率が大きくすなわち負荷が高い場合に前記コピーによりさらに負荷が生じることを回避することができ、また回避のための上限値をクラス600毎に任意に設定することができる。

【0068】上記のように処理することによって、ディスク装置502の使用状況に基づいて物理的に再配置する論理領域の選択、および再配置先の物理領域の選択を、クラス構成および属性に基づいて行い、再配置によりディスク装置502の負荷を分散して、各クラス600に設定されている使用率上限値を、クラス600に属するパリティグループ501の使用率が超えない配置を実現することができる。さらに再配置判断および実行の処理を繰り返して配置を修正していくことによって、使用状況の変動や予測誤差を吸収していくことができる。

【0069】再配置判断処理312において、制御部300は、対象期間の物理領域使用状況情報404のパリティグループ使用率や、論理領域使用状況情報403の論理領域のディスク装置使用率を参照して集計し、判断に用いるとしたが、例えば、対象期間の全ての値の平均を用いる代わりに、対象期間中の上位m個の値を用いる方法も考えられ、また上位m番目の値を用いる方法も考えられる（mは1以上の整数）。これらの方法をユーザが選択できるようにすることで、ユーザは使用状況の特徴的な部分のみを選択して用い、再配置判断処理312を行わせることができる。

【0070】上記の再配置判断処理312において、制御部300は、ディスクアレイシステム201の全てのクラス600について、論理領域の再配置の必要なパリティグループ501の検出を行うとしたが、前記検出の前に制御部300がクラス属性情報402を参照し、固定属性が設定されているクラス600については、検出の対象外としてもよい。また同様に、制御部300がパリティグループ情報409を参照し、固定属性が設定されているパリティグループ501については検出の対象外としてもよい。また、再配置判断処理312において、制御部300は、高性能クラスに属するパリティグループ501の未使用物理領域から再配置先の物理領域を選択するとしたが、固定属性が設定されているクラス600については対象外として、さらに性能順位が高位のクラス600を高性能クラスとして扱うようにしてもよい。また固定属性が設定されているパリティグループ501については対象外としてもよい。上記のように固

定属性が設定されているクラス６００またはパリティグループ５０１を設けることによって、ユーザは上記の自動的な再配置処理において物理的な再配置の影響を生じさせたくないクラス６００またはパリティグループ５０１を設定し、再配置の対象外とすることができる。

【００７１】＜第三の実施の形態＞本実施の形態では、同一クラス６００内での再配置判断について説明する。本実施の形態での計算機システムは、第２の実施の形態と同様である。ただし、本実施の形態では１つのクラス６００に複数のパリティグループ５０１が属する。本実施の形態での処理は、再配置判断処理３１２を除いては第２の実施の形態と同様である。また、再配置判断処理３１２についても、再配置する論理領域の選択（ステップ１６００）は、第２の実施の形態と同様である。

【００７２】本実施の形態での再配置判断処理３１２における、再配置先の物理領域の選択について図２０で説明する。

【００７３】第２の実施の形態では再配置先の物理領域を再配置元の物理領域の属するクラス６００より性能順位が高位のクラス６００から選択するが、本実施の形態では同一クラス６００の再配置元以外のパリティグループ５０１から選択する。制御部３００は、クラス構成情報４０１と未使用領域情報４０７を参照して、同一クラス６００に属する再配置元以外のパリティグループ５０１の未使用物理領域を取得する（ステップ１６１０）。制御部３００は、各未使用物理領域について、再配置先とした場合のパリティグループ使用率の予測値を求め（ステップ１６２０）、未使用物理領域の中から、再配置先とした場合に同一クラス６００に設定されている上限値を超えないと予測できる未使用物理領域を、再配置先の物理領域として選択し（ステップ１６３０）、選択結果を第２の実施の形態同様に、再配置情報４０８に出力する（ステップ１６４０）。再配置する全ての論理領域について再配置先の物理領域を選択し終えたら処理を終了する（ステップ１６５０）。

【００７４】上記の処理により、同一クラス６００内においてディスク装置５０２の負荷を分散することができる。上記の処理方法は例えばディスクアレイシステム２０１のパリティグループ５０１が全て１つのクラス６００（単一クラス）に属する構成に適用することができる。また、例えば、第２の実施の形態で説明した処理方法と組み合わせた場合に、再配置先の未使用物理領域の選択において、再配置元のクラス６００より性能順位が高位のクラス６００に適当な未使用物理領域が得られなかった場合や、性能順位が最上位のクラス６００での処理に適用できる。第２の実施の形態で説明した処理方法と組み合わせた場合は、第２の実施の形態での処理方法と本実施の形態での処理方法とが各クラス６００について異なる使用率上限値を用いてもよく、すなわち、そのためにクラス属性情報４０２が各クラス６００について二種類の使用率上限値または差分を有してもよい。

【００７５】＜第四の実施の形態＞本実施の形態では、第２の実施の形態での再配置判断処理３１２において、再配置元のクラス６００より性能順位が高位のクラス６００（高性能クラス）に再配置先の未使用物理領域が見つからなかった場合に、再配置先を得るために先立って行われる、性能順位がより低位のクラス６００（低性能クラス）への高性能クラスからの再配置の処理について説明する。

【００７６】本実施の形態での計算機システムは、第２の実施の形態と同様である。本実施の形態における再配置判断処理３１２について図２１で説明する。

【００７７】制御部３００は、高性能クラスに属するパリティグループ５０１をクラス構成情報４０１から取得する（ステップ１７００）。続いて制御部３００は、第１の実施の形態と同様の再配置判断対象期間情報４０５を参照して対象期間を取得し（ステップ１７１０）、対象期間の論理領域使用状況情報４０３を参照して、パリティグループ５０１の各物理領域に対応する論理領域のディスク装置使用率を取得し（ステップ１７２０）、ディスク装置使用率の小さいものから、低性能クラスへ再配置する論理領域として選択する（ステップ１７３０）。このとき論理領域の選択は必要なだけ行われる（ステップ１７４０）。

【００７８】続いて制御部３００は、選択された論理領域についての再配置先となる物理領域を、低性能クラスに属するパリティグループ５０１から選択するが、再配置先の物理領域選択の処理は、第２の実施の形態での処理説明において再配置先としている高性能クラスを低性能クラスと読み替えれば、第２の実施の形態での処理と同様である（ステップ１７５０）。また、本実施の形態におけるその他の処理も第２の実施の形態での処理と同様である。

【００７９】上記の処理を行うことで、第２の実施の形態での再配置判断処理３１２において高性能クラスに再配置先の未使用物理領域が見つからなかった場合に、高性能クラスから低性能クラスへ論理領域の再配置を、高性能クラスへの再配置に先立って行い、再配置先の未使用物理領域を高性能クラスに用意することができる。制御部３００は、上記の処理を必要に応じ繰り返し行って、十分な未使用物理領域を用意することができる。

【００８０】論理領域の再配置先を低性能クラスのパリティグループ５０１とするため、同一負荷に対するディスク使用時間が再配置について増大し、論理領域の再配置後のディスク装置使用率が増大する可能性があるが、ディスク使用率の小さい論理領域から再配置していくようにすることで、増大の影響を最小限に抑えることができる。

【００８１】＜第五の実施の形態＞本実施の形態では、クラス６００の属性の１つにアクセス種別属性を設け、

アクセス種別属性を用いてシーケンシャルアクセスが顕著に行われる論理領域とランダムアクセスが顕著に行われる論理領域とを、他のパリティグループ５０１に自動的に物理的再配置して分離するための再配置判断について説明する。

【００８２】本実施の形態における計算機システムは図１０に示したものである。本実施の形態では、第２の実施の形態での説明に加え、制御部３００が保持する下記の情報を用いる。

【００８３】本実施の形態でのクラス属性情報４０２の一例を図２２に示す。この例では、第２の実施の形態での例に対しアクセス種別が加えられており、クラス６００のアクセス種別が、例えばシーケンシャルに設定されている場合は、クラス６００がシーケンシャルアクセスに好適であると設定されていることを示す。

【００８４】本実施の形態での論理領域使用状況情報４０３の一例を図２３に示す。この例では、第２の実施の形態での例に対し、シーケンシャルアクセス率およびランダムアクセス率が加えられている。

【００８５】さらに、本実施の形態において制御部３００は、第２の実施の形態に加え、アクセス種別基準値情報４１０と論理領域属性情報４１１を保持する。

【００８６】アクセス種別基準値情報４１０の一例を図２４に示す。ユーザによりまたは初期条件として、アクセス種別基準値情報４１０には後述のアクセス種別の判定に用いる基準値が設定されている。また、論理領域属性情報４１１の一例を図２５に示す。アクセス種別ヒントは、各論理領域について顕著に行われると期待できるアクセス種別であり、ユーザが設定する。固定については後述する。

【００８７】本実施の形態での処理は、使用状況情報取得処理３１１および再配置判断処理３１２を除いては第二の実施の形態と同様である。

【００８８】本実施の形態における使用状況情報取得処理３１１について図２６で説明する。

【００８９】制御部３００は、第２の実施の形態での使用状況情報取得処理３１１と同様に、論理領域についてのディスク装置使用率を算出し（ステップ１８００、１８１０）、リード／ライト処理３１０での使用率内容を分析して、使用率についてシーケンシャルアクセスとランダムアクセスの比率を算出し（ステップ１８２０）、使用率およびアクセス種別比率を論理領域使用状況情報４０３に記録する（ステップ１８３０）。また、制御部３００は、第２の実施の形態と同様にパリティグループ使用率の算出と物理領域使用状況情報４０４への記録を行う（ステップ１８４０、１８５０）。

【００９０】本実施の形態における再配置判断処理３１２において、再配置する論理領域の選択は第２の実施の形態と同様である（ステップ１９９０）。再配置判断処理３１２での再配置先の物理領域の選択について図２７で説明する。

【００９１】制御部３００は、論理領域使用情報４０３を参照し、再配置する論理領域についてのシーケンシャルアクセス率を取得し（ステップ１９１０）、アクセス種別基準値情報４１０に設定されている基準値と比較する（ステップ１９２０）。シーケンシャルアクセス率が基準値より大きい場合、制御部３００は、クラス属性情報４０２を参照し、アクセス種別がシーケンシャルと設定されているクラス６００（シーケンシャルクラス）が存在するか調べる（ステップ１９５０）。シーケンシャルクラスが存在する場合、制御部３００は、クラス構成情報４０１と未使用領域情報４０７を参照して、シーケンシャルクラスに属する再配置元以外のパリティグループ５０１の未使用物理領域を取得する（ステップ１９６０）。さらに制御部３００は、各未使用物理領域について、再配置先とした場合のパリティグループ使用率の予測値を求め（ステップ１９７０）、未使用物理領域の中から、再配置先とした場合にシーケンシャルクラスに設定されている上限値を超えないと予測できる未使用物理領域を、再配置先の物理領域として選択し（ステップ１９８０）、選択結果を第２の実施の形態同様に再配置情報４０８に出力する（ステップ１９９０）。制御部３００は、使用率予測値を、第２の実施の形態と同様のパリティグループ情報４０９と本実施の形態における論理領域使用状況情報４０３および物理領域使用状況情報４０４から算出する。

【００９２】前記の比較において、シーケンシャルアクセス率が基準値以下である場合、制御部３００は、論理領域属性情報４１１を参照し、論理領域についてアクセス種別ヒントがシーケンシャルと設定されているか調べる（ステップ１９４０）。アクセス種別ヒントにシーケンシャルと設定されていた場合、上記と同様に制御部３００は、シーケンシャルクラスの有無を調べ（ステップ１９５０）、シーケンシャルクラスが存在する場合は、シーケンシャルクラスから再配置先の物理領域を選択する（ステップ１９６０～１９９０）。

【００９３】前記の比較において、シーケンシャルアクセス率が前記基準値以下であり、さらにアクセス種別ヒントがシーケンシャルでなかった場合、またはシーケンシャルクラスが存在しなかった場合、制御部３００は、第２の実施の形態と同様に、シーケンシャルクラス以外のクラス６００から再配置先の物理領域を選択する（ステップ２０００）。

【００９４】上記の処理により、同一パリティグループ５０１での顕著なシーケンシャルアクセスとランダムアクセスの混在に対し、各クラス６００に属性として設定されたアクセス種別と使用率上限値を用いて、シーケンシャルアクセスが顕著に行われる論理領域とランダムアクセスが顕著に行われる論理領域とを、異なるパリティグループ５０１に自動的に再配置して分離、すなわち異

なるディスク装置５０２に分離することができ、特にランダムアクセスに対する応答性能を改善することができる。

【００９５】また、上記の処理においては制御部３００は、シーケンシャルアクセスに注目して再配置による自動的分離を行うとしたが、同様にランダムアクセスに注目して前記分離を行うことも可能である。

【００９６】上記の再配置判断処理３１２において、再配置する論理領域を選択した時点で、制御部３００が論理領域属性情報４１１を参照し、論理領域に固定属性が指定されている場合は、論理領域を再配置しないとすれば、ユーザが特に再配置を行いたくないと考える論理領域がある場合、固定属性を設定することで論理領域を再配置の対象外とすることができる。上記の固定属性に関する処理は論理領域属性情報４１１を用いることで、前述の実施の形態にも適用できる。

【００９７】

【発明の効果】ストレージサブシステムのユーザ、または保守員が、記憶領域の物理的再配置による配置最適化を行うための作業を簡便にすることができる。

【図面の簡単な説明】

【図１】本発明の第１の実施の形態での計算機システムの構成図である。

【図２】本発明の第１の実施の形態でのリード／ライト処理３１０および使用状況情報取得処理３１１のフローチャートである。

【図３】本発明の第１の実施の形態での論理／物理対応情報４００の一例を示す図である。

【図４】本発明の第１の実施の形態での再配置判断処理３１２のフローチャートである。

【図５】本発明の第１の実施の形態での再配置判断対象期間情報４０５の一例を示す図である。

【図６】本発明の第１の実施の形態での再配置情報４０８の一例を示す図である。

【図７】本発明の第１の実施の形態での未使用領域情報４０７の一例を示す図である。

【図８】本発明の第１の実施の形態での再配置実行処理３１３のフローチャートである。

【図９】本発明の第１の実施の形態での再配置実行時刻情報４０６の一例を示す図である。

【図１０】本発明の第２の実施の形態および第五の実施の形態の計算機システムの構成図である。

【図１１】本発明の第２の実施の形態での論理／物理対応情報４００の一例を示す図である。

【図１２】本発明の第２の実施の形態でのクラス構成情報４０１の一例を示す図である。

【図１３】本発明の第２の実施の形態でのクラス属性情報４０２の一例を示す図である。

【図１４】本発明の第２の実施の形態での使用状況情報取得処理３１１のフローチャートである。

【図１５】本発明の第２の実施の形態での論理領域使用状況情報４０３の一例を示す図である。

【図１６】本発明の第２の実施の形態での物理領域使用状況情報４０４の一例を示す図である。

【図１７】本発明の第２の実施の形態での再配置判断処理３１２のフローチャートである。

【図１８】本発明の第２の実施の形態でのパリティグループ情報４０９の一例を示す図である。

【図１９】本発明の第２の実施の形態での再配置実行処理３１３のフローチャートである。

【図２０】本発明の第３の実施の形態での再配置判断処理３１２のフローチャートである。

【図２１】本発明の第４の実施の形態での再配置判断処理３１２のフローチャートである。

【図２２】本発明の第５の実施の形態でのクラス属性情報４０２の一例を示す図である。

【図２３】本発明の第５の実施の形態での論理領域使用状況情報４０３の一例を示す図である。

【図２４】本発明の第５の実施の形態でのアクセス種別基準値情報４１０の一例を示す図である。

【図２５】本発明の第５の実施の形態での論理領域属性情報４１１の一例を示す図である。

【図２６】本発明の第５の実施の形態での使用状況情報取得処理３１１のフローチャートである。

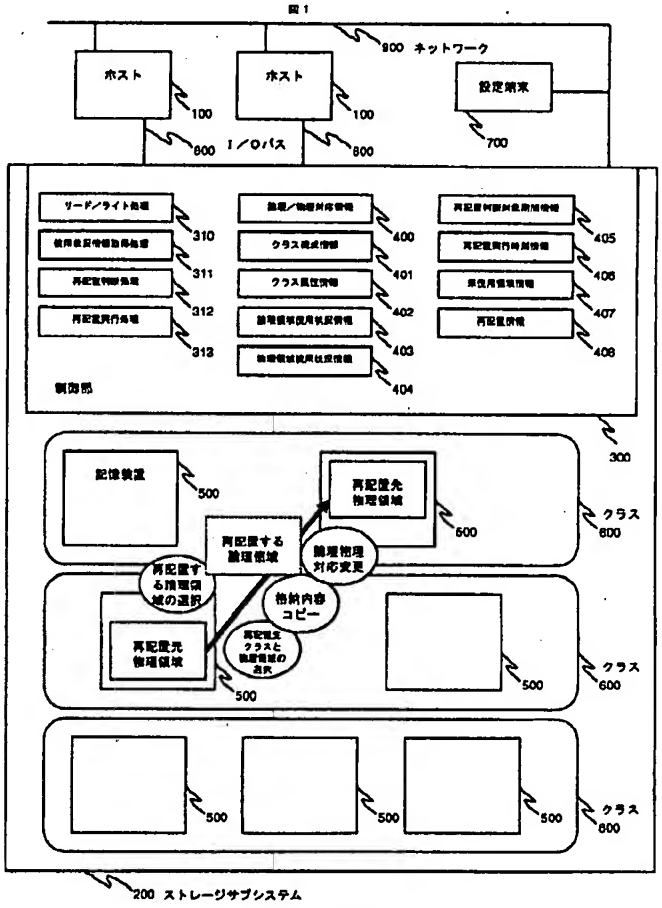
【図２７】本発明の第５の実施の形態での再配置判断処理３１２のフローチャートである。

【符号の説明】

１００　ホスト
２００　ストレージサブシステム
２０１　ディスクアレイシステム
３００　制御部
３１０　リード／ライト処理
３１１　使用状況情報取得処理
３１２　再配置判断処理
３１３　再配置実行処理
４００　論理／物理対応情報
４０１　クラス構成情報
４０２　クラス属性情報
４０３　論理領域使用状況情報
４０４　物理領域使用状況情報
４０５　再配置判断対象期間情報
４０６　再配置実行時刻情報
４０７　未使用領域情報
４０８　再配置情報
４０９　パリティグループ情報
４１０　アクセス種別基準値情報
４１１　論理領域属性情報
５００　記憶装置
５０１　パリティグループ
５０２　ディスク装置

６００　クラス
７００　制御端末
８００　Ｉ／Ｏパス
９００　ネットワーク

【図１】



【図９】

図９

日時	１９９９年８月１１日　２２時０分
日時更新情報	毎日（＋２４時間）

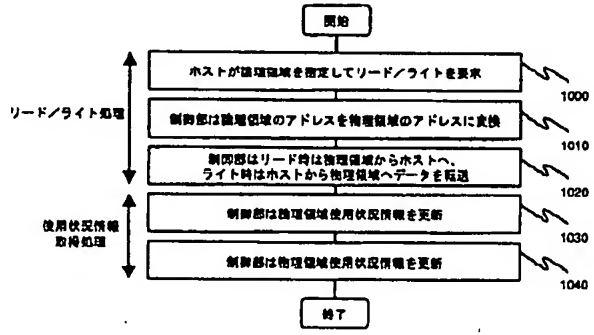
【図２４】

図２４

アクセス判定基準値（%）
７５

【図2】

図2



【図3】

図3

論理アドレス	物理アドレス	
	記憶装置番号	記憶装置内アドレス
0~999	0	0~999
1000~1999	0	1000~1999
2000~2999	1	0~999
3000~3999	1	1000~1999

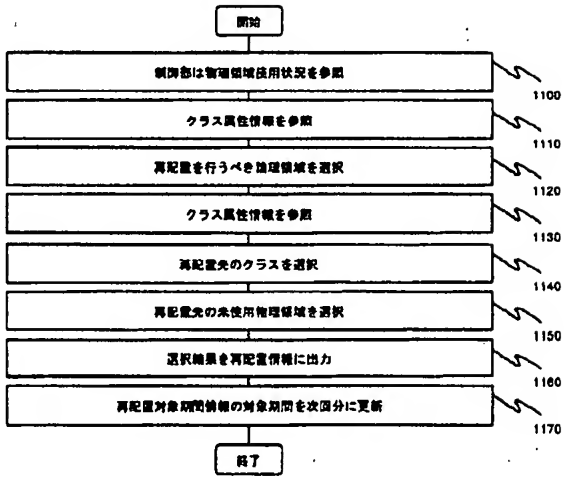
【図5】

図5

開始日時	1999年8月11日 8時30分
終了日時	1999年8月11日 17時15分
期間更新情報	毎日（+24時間）

【図4】

図4



【図6】

図6

番号	論理領域	再配置元物理領域		再配置先物理領域	
		記憶装置番号	記憶装置内アドレス	記憶装置番号	記憶装置内アドレス
1	0~999	0	0~999	10	0~999
2	1000~1999	0	1000~1999	10	1000~1999

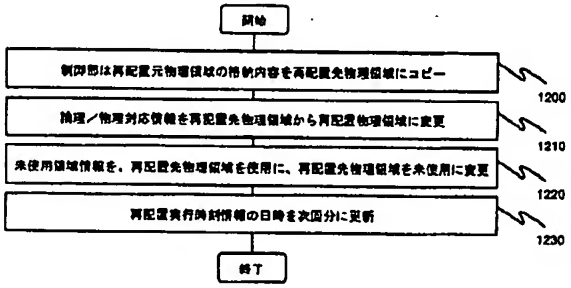
【図7】

図7

記憶装置番号	記憶装置内アドレス	使用/未使用
0	0~999	使用
0	1000~1999	使用
0	2000~2999	未使用
0	3000~3999	未使用

【図8】

図8



【図11】

図11

論理アドレス	物理アドレス				
	パリティグループ番号	データ		冗長データ	
		記憶装置番号	記憶装置内アドレス	記憶装置番号	記憶装置内アドレス
0～999	100	0	0～999	20	0～999
1000～1999	100	0	1000～1999	20	1000～1999
2000～2999	101	1	0～999	41	0～999
3000～3999	101	1	1000～1999	41	1000～1999

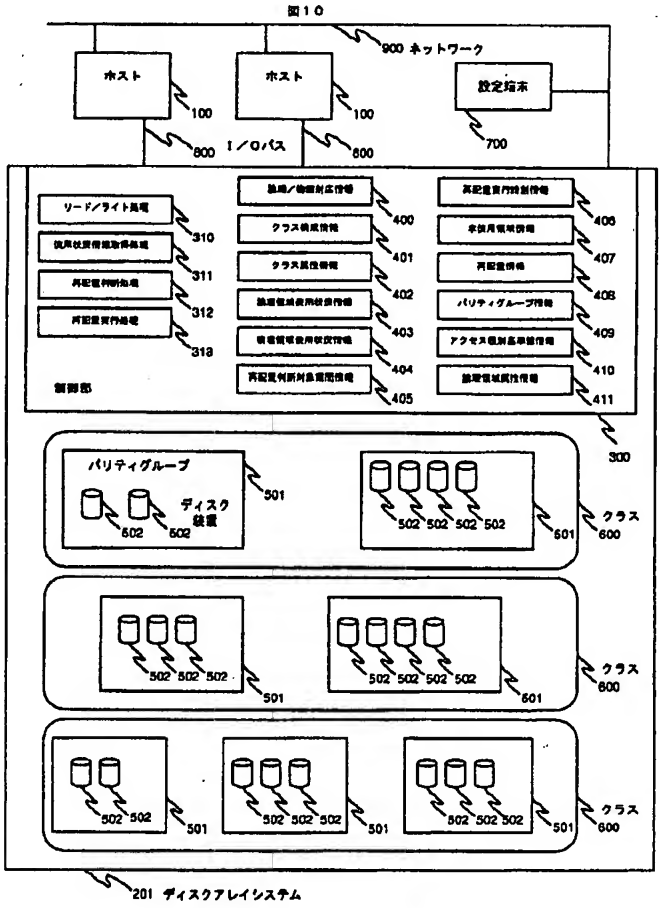
【図12】

図12

クラス番号	パリティグループ数	パリティグループ番号
0	3	100, 110, 120
1	2	101, 111
2	4	102, 112, 122, 132

【図10】

図10



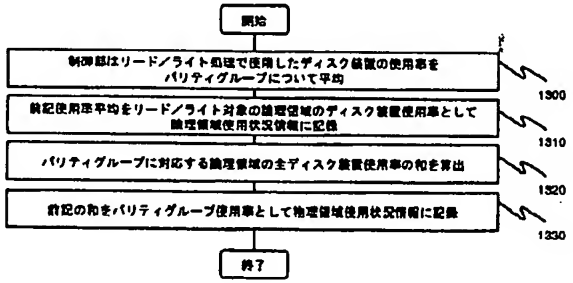
【図13】

図13

クラス番号	使用率上限値（%）	クラス間性能順位	再配置実行上限値（%）	固定
0	60	1	70	−
1	70	2	80	固定
2	80	3	90	−

【図14】

図14



【図15】

図15

日時	論理アドレス	ディスク装置使用率（%）
1999年8月11日 8時0分	0〜999	18
	1000〜1999	32
1999年8月11日 8時15分	0〜999	20
	1000〜1999	30
1999年8月11日 8時30分	0〜999	22
	1000〜1999	28

【図16】

図16

日時	パリティグループ番号	使用率（%）
1999年8月11日 8時0分	100	68
	101	52
1999年8月11日 8時15分	100	70
	101	50
1999年8月11日 8時30分	100	72
	101	46

【図18】

図18

パリティグループ番号	RAID構成	ディスク装置性能	固定
100	RAID5 3D1P	110	−
101	RAID1 1D1P	100	固定
102	RAID5 6D1P	95	−

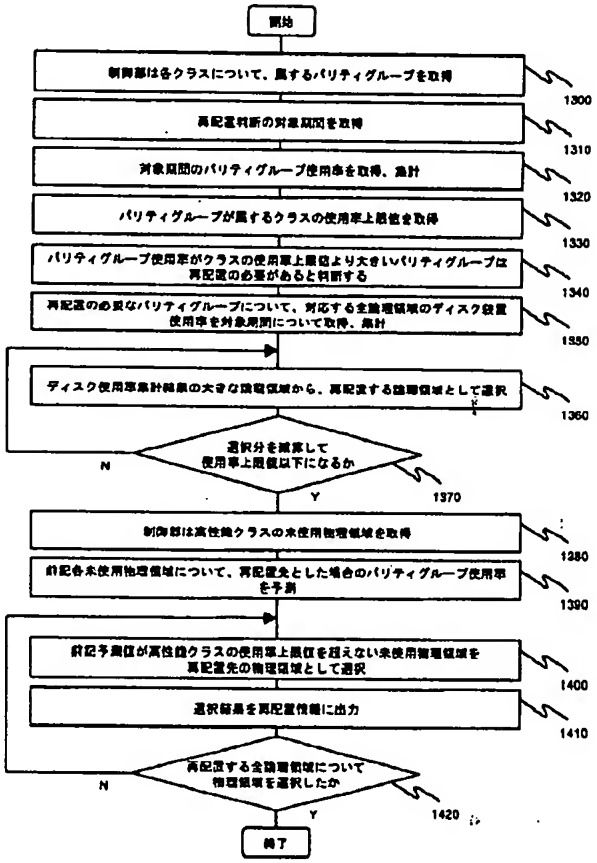
【図22】

図22

クラス番号	使用率上限値（%）	クラス間性能順位	再配置実行上限値（%）	固定	アクセス種別
0	60	1	70	−	−
1	70	2	80	−	−
2	80	3	90	−	シーケンシャル

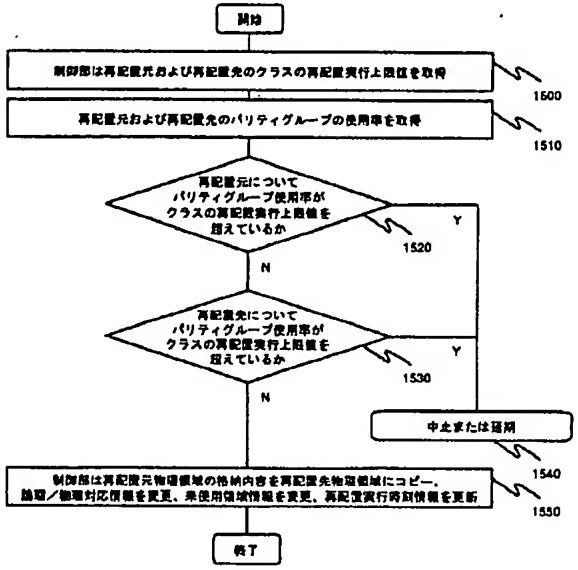
【図１７】

図１７



【図１９】

図１９



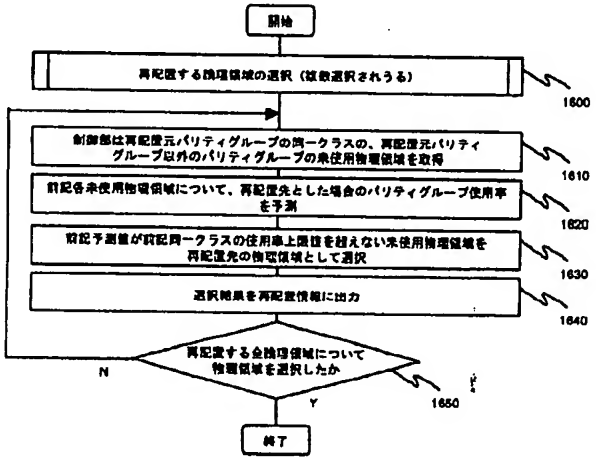
【図２３】

図２３

日時	論理アドレス	ディスク装置使用率（%）	シーケンシャルアクセス率（%）	ランダムアクセス率（%）
１９９９年８月１１日　８時０分	０～９９９	１８	７８	２２
	１０００～１９９９	３２	５２	４８
１９９９年８月１１日　８時１５分	０～９９９	２０	８０	２０
	１０００～１９９９	３０	５０	５０
１９９９年８月１１日　８時３０分	０～９９９	２２	８２	１８
	１０００～１９９９	２８	４８	５２

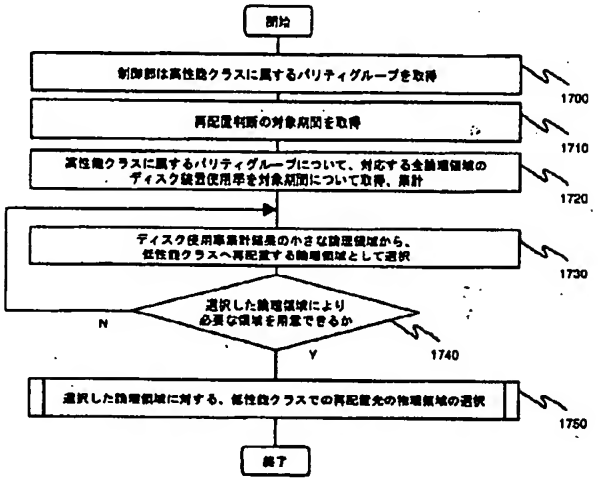
【図20】

図20



【図21】

図21



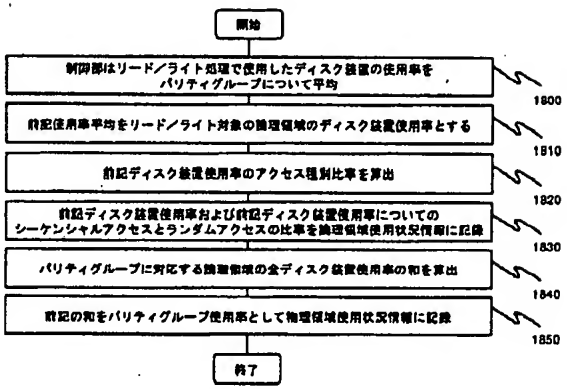
【図25】

図25

論理アドレス	アクセス種別ヒント	固定
0～999	–	–
1000～1999	–	–
2000～2999	シーケンシャル	–
3000～3999	–	固定

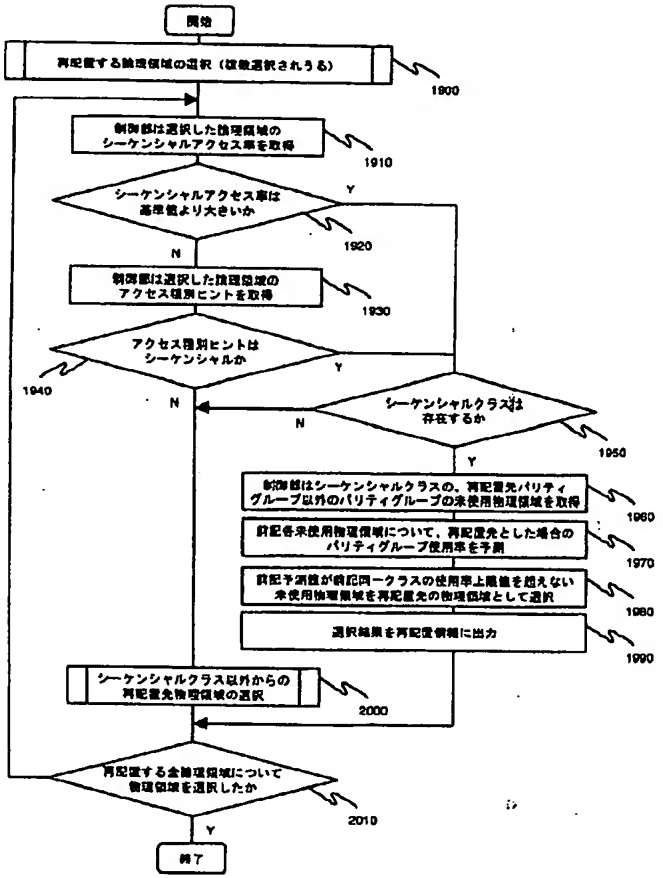
【図26】

図26



【図２７】

図２７



フロントページの続き

(72)発明者　山神　恵司
神奈川県川崎市麻生区王禅寺1099番地　株式会社日立製作所システム開発研究所内

(72)発明者　荒井　弘治
神奈川県小田原市国府津2880番地　株式会社日立製作所ストレージシステム事業部内

Ｆターム(参考)　5B065 BA01 CA30 CC01 CC03 EK01
5B082 CA11

3B-3

高機能ディスクにおけるアクセスプランを用いたプリフェッチ機構に関する評価

向井 景洋, 根本 利弘, 喜連川 優

東京大学 生産技術研究所

〒106-8558 東京都港区六本木 7-22-1

Tel 03-3402-6231 Fax 03-3479-1706

E-mail: {mukai,nemoto,kitsure}@tkl.iis.u-tokyo.ac.jp

概要

半導体技術の進歩により、プロセッサ、メモリは急速に高性能化しているにも関わらず、コストは急激に下がってきている。このような状況を背景に、単なる容量としてのディスクではなく、高性能のコントローラや大容量のメモリを搭載し、複雑な処理を行うことの出来るディスクに注目が集まっている。現在、様々な研究機関で、インテリジェントなディスクに関する研究が行われている。これらの研究で提案されているディスクは、データベースアプリケーションの性能を大幅に向上させる一方で、ホストのコードの大幅な変更を必要とする。そこで、本論文では、アプリケーションレベルの知識を理解し、効率的なI/O処理を行う、インテリジェントなディスクを提案する。このようなディスクでは、ホストコードの変更が少ないことが特徴として挙げられる。このディスクに於いて、アクセスプランを用いたプリフェッチを行った場合に、どの程度の性能向上が見込まれるのかを調べる為に、シミュレーション実験を行った。その結果、実行時間の大幅な削減が期待できることが判った。

Evaluation of Prefetching Mechanism Using Access Plan on Intelligent Disk

Kazuhiro Mukai, Toshihiro Nemoto and Masaru Kitsuregawa

Institute of Industiral Science, The University of Tokyo
7-22-1, Roppongi, Minato-ku, Tokyo 106-8558, Japan

Abstract

The progress of semiconductor technologies has made high performance processors and memory chips available at cheaper price. Therefore, it has become possible for disks to process complicated tasks. There are several research projects in the field of intelligent disk architecture. Although they improve the performance of database applications significantly, dbms code has to be modified a lot. In this paper, we examine the access plan based prefetching approach where the access plan is given to the disk controller. This approach needs very little change in host code. We examine how our access plan based prefetching approach improves the performance in detail. Experiment results indicate that the execution time decreases dramatically.

1 はじめに

半導体技術の進歩により、プロセッサ、メモリは急速に高性能化しているにも関わらず、そのコストは急激に下がってきている。現在のハイエンドなディスクのコントローラには100MHzのプロセッサが用いられ、ディスク・キャッシュのサイズはすでに4MB(最大16MB)[8]に達している。そのため、ロボットアームなどの物理的な動きを必要とするシーク時間や回転待ち時間が数ミリ秒であるのに対して、0.1ミリ秒のオーダーで処理を終えるコントローラの性能が際だっている。一方、データベースの世界では、DSSやデータ・ウェアハウスは急速に大規模化し、システムにおける処理性能の向上や容量の増加が強く要求されている。データの急速な増加に対応する為に、ユーザは膨大な量のデータを複数のディスクに保存する。大量なデータを多数のディスクに保持し、並列にアクセスするシステムにおいては、ホストのコンピュータはディスク間の調整に大部分の時間を割くことが必要となる。また、ホストとディスクを繋ぐネットワークのバンド幅はボトルネックになりやすい。

以上で述べた事を背景に、単なる容量としてのディスクではなく、高性能のコントローラや大容量のメモリを搭載し、複雑な処理を行うことの出来るインテリジェントなディスクに注目が集まっている。

本論文では、アプリケーションレベルの知識を理解し、それを基に効率の良いI/O処理を行うインテリジェントなディスクを提案する。このようなディスクでは、アプリケーションレベルの知識を理解できる為、負荷が比較的軽い時間を利用し、将来、ホストが参照すると予想されるデータを効率の良いスケジューリングのもとでプリフェッチを行っておくことにより、キャッシュヒット率を向上させることができ、性能の向上が期待できる。

インテリジェントなディスクに関する研究例としては、データベースにおけるSELECT、GROUP BY、外部ソート、データキューブ、画像処理などのアプリケーションに関し、ディスクに内在しストリームベースでデータを処理するディスクレットを用いた研究[3]、大規模なSMPシステムにおいて、スキャン、ハッシュジョイン、ソートをディスクで行う事により、ホストへのデータ転送を削減する研究[4]、近似データサーチ、データマイニング、画像処理のスキャンベースのアルゴリズムを扱った研究[2]、などが提案されている。これらの研究では、複雑な処理をディスク内で行う為、ホストのコードの大幅な変更を必要とする。これに対し、本論文で提案する方式では、ホスト側の変更は少なく、既存のシステムに適用しやすい事が利点として挙げられる。

本論文では、データベースに於けるアクセスプランが与えられたディスクにおいて、そのアクセスプランを用い、知的なアクセススケジューリングによるプリフェッチが行われた場合、どの程度の性能の向上が見込まれるのかを評価する為に、データベースのベンチマークであるTPC-Hをアプリケーションとして用い、シミュレーション実験を行った。その結果、大きな実行時間の削減が期待できるという結果が得られた。

以下、2章で従来のディスクアクセスについて簡単に説明し、問題点を明らかにする。その後、アクセスプランを用いたプリフェッチ機構について説明をする。3章では2章で提案したプリフェッチ機構に関する評価実験の詳細を述べる。4章では評価実験の結果を示し、考察を行う。最後に、5章で、まとめと今後の課題を述べる。

2 アクセスプランを用いたディスクプリフェッチ

2.1 従来のディスクアクセスとその問題点

アプリケーションの例として、データベースのベンチマークの一つであるTPC-Hを考える。まず、TPC-Hの問合せの一つであるQuery8のSQLとORACLE8によって作成されるアクセスプランの一部を図1に示す。

図1のアクセスプランに示されるように、Q8で、ホストは、where句に関して、PARTS,LINEITEM,ORDERSのインデックス、またはテーブルにアクセスし、Nested Loops Joinを行っている。図1に示したSQL文に関するORACLEシステムのディスクアクセスについて詳しく検討していく。

まず、ORACLEシステムは、

$$P\_TYPE = \text{'ECONOMY ANODIZED STEEL'} \quad (1)$$

の条件を満たすレコード番号を得る為に、PARTSのインデックスにアクセスを行う。その後、

$$P\_PARTKEY = L\_PARTKEY \quad (2)$$

の条件により、式(1)より得られたレコード番号のPARTSテーブル、LINEITEMのインデックスにアクセスし、NESTED LOOPS JOINを行う。次に、

$$L\_ORDERKEY = O\_ORDERKEY \quad (3)$$

と

$$O\_ORDERDATE \quad (4)$$

の条件を満たすレコードを見つける為に、ORDERSのインデックス、テーブルに順次アクセスを行い、式(2)で得られたレコードとNESTED LOOPS JOINを行っている。L_ORDERKEYに関しては、式(2)で

```
SELECT YEAR,
       SUM(DECODE(NATION,'BRAZIL',VOLUME,0))/SUM(VOLUME)AS MKT_SHARE
FROM ( SELECT TO_CHAR ( O_ORDERDATE, 'YYYY' ) AS YEAR,
       L_EXTENDEDPRICE * ( 1-L_DISCOUNT ) AS VOLUME, N2.N_NAME AS NATION
   FROM PARTS, SUPPLIER, LINEITEM, ORDERS, CUSTOMER, NATION N1,
        NATION N2, REGION
   WHERE P_PARTKEY = L_PARTKEY
   AND P_TYPE = 'ECONOMY ANODIZED STEEL'
   AND L_ORDERKEY = O_ORDERKEY
   AND O_ORDERDATE BETWEEN TO_DATE('1995-01-01','YYYY-MM-DD')
   AND TO_DATE('1996-12-31','YYYY-MM-DD')
   AND S_SUPPKEY = L_SUPPKEY                AND O_CUSTKEY = C_CUSTKEY
   AND C_NATIONKEY = N1.N_NATIONKEY         AND N1.N_REGIONKEY = R_REGIONKEY
   AND R_NAME = 'AMERICA'                   AND S_NATIONKEY = N2.N_NATIONKEY )
   AS ALL_NATIONS
GROUP BY YEAR
ORDER BY YEAR;
```

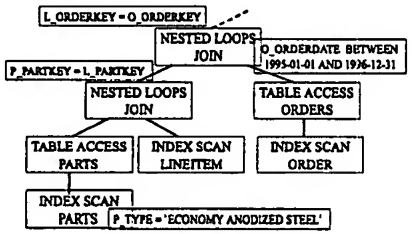


図 1: (左)Query8 の SQL、(右)Query8 のアクセスプランの一部

読み込んだ LINEITEM のインデックス内にデータが存在していた為、テーブルにはアクセスしていない。

Q8 の ORACLE8 による実際のディスクアクセスの様子を、図 2に示す。横軸は実行時間、縦軸は LBA(論理ブロックアドレス) を示している。この図に於いて、プロットした点と点を結ぶ線が、ディスクのシークを表していると考えることが出来る。

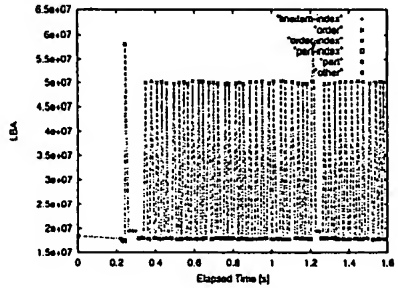


図 2: Q8 のディスクアクセス (最初の 1.6ms)

図 2からも分かるように、Q8 に関し、ディスクは、ORDERS のインデックスとテーブルに交互にアクセスする為、ディスクアクセス時間の支配的要素の一つであるシーク時間の点から考えると、非常に効率の悪いアクセスが行われていると考えられる。また、ホストは、過去にアクセスしたデータに基づき、ディスクアクセスを行う為、ホストがデータを受け取って、次の命令を出すまでの間、ディスクはストールした状態となっている。(図 3)

2.2 アクセスプランベースのプリフェッチ

次に、アプリケーションレベルの知識として、アクセスプランがディスクに与えられた場合を考える。この場合、ディスクは、ORDERS のインデックスを読んだ後、ORDERS のテーブルのどのブロックにアクセスすべきなのか判断できるようになる。従って、ORDERS のインデックスを連続的にアクセスし、そのインデックスによりアクセスすべきテーブルのブロック群をスケジューリングし、物理的に近い所からまとめてアクセスする事により、シーク時間を短縮する事が出来る。また、このディスクアクセスは、ホストからの命令を待つ事なく行われる為、ホストとディスクは、並行して処理作業を行う事ができる。(図 4)
さらに、データがディスクアレイのような複数台のディスクに分散して保存されている場合、個々のディスクに並行してアクセスできる為、さらなる実行時間の短縮が期待できる。

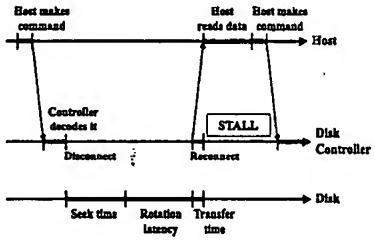


図 3: ディスクのストール

3 実ディスクアレイを用いた擬似実験

ディスクにおいてアクセスプラン情報を用いたスケジューリングによるプリフェッチを行った場合、どの程度の性能向上が見込まれるのかを評価する為に、ディスクコントローラによるプリフェッチを擬似的に実現するシステムを構築し、評価実験を行う。本来、ディスクコントローラを高機能化し、プリフェッチ機構はディスク上に実装されるべきものであるが、その構築は困難であることから、ホスト上にプリフェッチ機構を擬似的に実行するプリフェッチプロセスを作成し、代行させる。

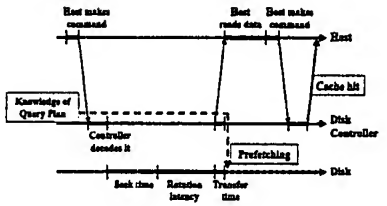


図 4: アクセスプランを用いたディスクアクセス

3.1 実験環境

実験環境を図 5に示す。SUN Ultra Enterprize450(4 × Ultra SPARC-II 296MHz) をホストとし、ディスクアレイ (Hitachi DF400) を接続する。ディスクアレイにはアクセス用のポートが 2 つあり、それぞれがホストに Wide SCSI インタフェースでつながれている。DBMS において問合せを実行した場合、ディスクアレイには 1 方のポートからアクセスが行われる。ここで、DBMS がアクセスするポートを DBMS ポートと名付け、プリフェッチプロセスがアクセスするポートを prefetch ポートと名付ける。ORACLE 用の TPC-H のデータサイズは Scale Factor = 2、すなわち約 2G(Byte) で、ディスクアレイのローデバイス上に RAID5(4D+1P) で保存されている。ホストは、この 2 つのポートに対し、個別にコマンドを発行する事ができる。また、ディスクのキャッシュは 2 つのポートに於いて共有されている。実験に使用した TPC-H の各テーブルの構成を表 1に示す。

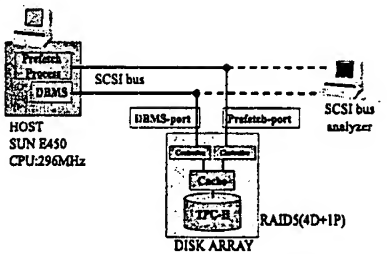


図 5: シミュレーション環境

また、ディスクアレイのそれぞれのポートへのアクセス状況は、SCSI バスアナライザを通し、PC により採取することが出来る。

	レコード数	block 数	size(Byte)
PARTS	400K	28570	59M
LINEITEM	12,000K	755350	1,546M
ORDERS	3,000K	183820	376M
PARTSUPP	1600K	125910	258M
SUPPLIER	20K	1590	3M
CUSTOMER	300K	26160	52M
NATION	25	2	4k
REGION	5	1	2k

表 1: 実験用 TPC-H データセット

	Q8
PARTS.INDEX	56
PARTS	2711
LINEITEM.INDEX	6562
ORDERS.INDEX	41003
ORDERS	68164
OTHER	4913
TOTAL	123409

表 2: 各テーブルへのアクセス数 (TPCH Q8)

3.2 ディスクアクセストレース

TPC-H において、ORACLE システムが Q8 を実行した場合のアクセストレースを、SCSI バスアナライザを用い採取した。Q8 の実行時間は、1180 秒であった。また、Q8 に関して行われた各テーブルへのアクセス数を表 2に示す。ORACLE システムの最小アクセスブロック単位は 2k(Byte) であり、ほとんどのアクセスが最小ブロック単位で行われていた。

3.3 プリフェッチシステム

この節では、ディスクコントローラによるプリフェッチを擬似的に実現するプリフェッチシステムについて、説明を行う。

ORACLE8 は DBMS ポートを通し、ディスクアクセスを行う。この時、修正されたデバイスドライバにより DBMS ポートからの全てのディスクアクセス情報が、アクセスが発生すると同時にプリフェッチプロセスに知らされる。プリフェッチプロセスは、このアクセスを監視し、ORACLE が特定のブロックへアクセスを行うと、トレース情報を基に、prefetch ポートを通して、将来、ORACLE が参照すると考えられる

データのプリフェッチを実行する。ディスクアレイ上では、キャッシュが2つのポート間で共有されているため、ディスクコントローラがプリフェッチを行った場合と同じ状況が擬似的に作成される。プリフェッチシステムを図6に示す。

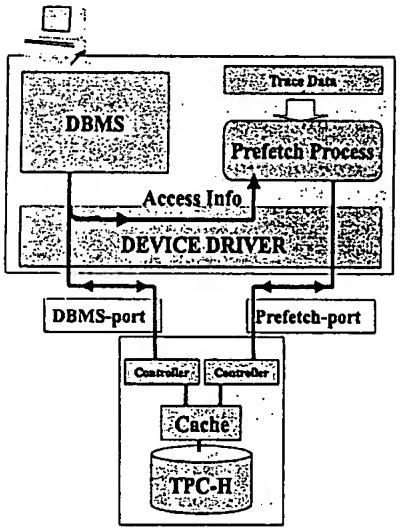


図 6: プリフェッチシステム

3.4 プリフェッチアルゴリズム

この節では、プリフェッチのアルゴリズムについて説明する。アクセスプランに於いて、NESTED LOOPS JOIN が行われている部分の最内周に注目した時、その最内周がインデックスを用いたテーブルアクセスである場合、その最内周の部分に於いて、アクセススケジューリングを用いたプリフェッチによるシーク時間の短縮の効果が期待できる。アクセスプランが多重の NESTED LOOPS JOIN になっている場合、効率的なスケジューリングを行うという観点からは、より外周側のループによるプリフェッチが望ましいが、一度の先読みで得たデータを ORACLE システムが参照するまでディスクキャッシュから溢れない様に利用する必要があり、ディスクキャッシュ容量の制限を受ける。よって、一度に行うプリフェッチの量(プリフェッチの深さ)は、統計情報などにより必要なディスクキャッシュ容量を予測し、決定される。

Q8 を例に取り、説明することにする。図 1のアクセスプランを考えた場合、Q8 は 2 重の NESTED LOOPS JOIN を構成している。最内周は、ORDERS.INDEX を用いた ORDERS へのアクセスであり、LINEITEM.INDEX、PARTS、PARTS.INDEX の条件節の基で、アクセスされる。Q8 の場合の最も浅い先読みは ORACLE が LINEITEM.INDEX にアクセスを行ったと同時に、その LINEITEM.INDEX から得られる ORDERS.INDEX と ORDERS のプリフェッチを行うことである。Q8 の場合のプリフェッチのループの候補は、

1. ORDERS.INDEX,ORDERS のプリフェッチ (LINEITEM.INDEX ベース)
2. LINEITEM.INDEX,ORDERS.INDEX,ORDERS のプリフェッチ (PARTS ベース)
3. PARTS,LINEITEM.INDEX,ORDERS.INDEX, ORDERS のプリフェッチ (PARTS.INDEX ベース)
4. PARTS.INDEX,PARTS,LINEITEM.INDEX, ORDERS.INDEX,ORDERS のプリフェッチ

となる。ここで、番号が小さい程、浅いプリフェッチを示し、番号が大きい程、深いプリフェッチを示している。今回の実験では、Q8 に関して、LINEITEM.INDEX ベース (番号 1)、PARTS ベース (番号 2)、PARTS.INDEX ベース (番号 3) のループに関して実験を行った。

	実行時間 (s)
プリフェッチ無し	1180
LINEITEM.INDEX ベース	423
PART ベース	338
PART.INDEX ベース	233

表 3: Q8 における実行時間の比較 (同時先読み IO 発行数 30)

4 実験結果

4.1 プリフェッチの深さと実行時間

同時先読み IO 発行数が 30 の場合の LINEITEM.INDEX ベース、PARTS ベース、PARTS.INDEX ベースのプリフェッチを行った場合の実行時間を表 3に示す。表から明らかな様に、プリフェッチを行わない場合と比較して、プリフェッチを行った場合の実行時間が大きく短縮されていることが判る。また、プリフェッチを行った場合について比較すると、プリフェッチの深さを深くした場合の方がより大幅な性能の向上が示されている。最も深いプリフェッチでは、プリフェッチ無しの場合の 5 倍以上の性能向上が示されている。

以下、アクセストレースを見ながら、より詳細な解析を行う。

4.2 プリフェッチの深さ

この節では、プリフェッチの深さの影響について考える。同時先読み IO 発行数が 30 の場合の LINEITEM.INDEX ベース、PARTS ベース、PARTS.INDEX ベースの DBMS ポート、prefetch ポートからの I/O 発行トレースを図 7、図 8(LINEITEM.INDEX ベース)、図 9、図 10(PARTS ベース)、図 11、図 12(PARTS.INDEX ベース) に示す。横軸は時間を示し、縦軸は LBA を示している。

これらの図によって、ORACLE がベースとなるブロックへのアクセスを行ったと同時に、prefetch ポートよりプリフェッチが開始されていることが、確認できる。LINEITEM.INDEX ベースの場合、ORACLE が LINEITEM.INDEX へアクセスを行うと同時に、プリフェッチプロセスはその LINEITEM.INDEX から得られる ORDERS.INDEX と ORDERS のプリフェッチを行う。よって、そのブロック群がプリフェッチされた後、プリフェッチプロセスは、ORACLE が次の LINEITEM.INDEX にアクセスを行うまでストールする。図 8に於いて、棒線の部分がプリフェッチ機能のストールを示している。次に PARTS ベースの場合について考える。PARTS ベースのプリフェッチを行った場合、今度は、PARTS 毎にプリフェッチ機能のストールが起こる (図 10) が、LINEITEM.INDEX ベースの場合 (図 8) と比べると、はるかに少ないことが判る。PARTS.INDEX ベースの場合には、PARTS.INDEX 毎にストールするが、そのストールは PARTS ベースの時よりもさらに少なく、図 12の枠に入る時間内には一度も発生しない。この様に先読みが深い程、プリフェッチ機能がストールする場合が少なくなり、性能が向上する事になる。

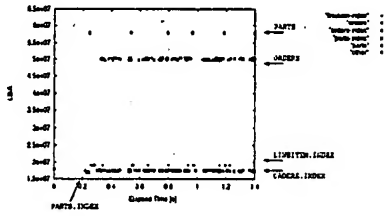


図 7: DBMS ポートのアクセス (LINEITEM.INDEX ベース)

4.3 キャッシュヒット状況

プリフェッチの深さごとのアクセスのヒット状況について検証する。

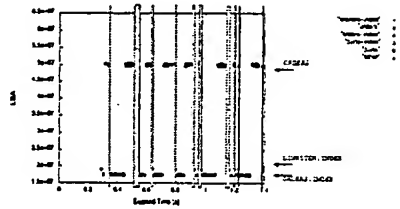


図 8: prefetch ポートのアクセス (LINEITEM.INDEX ベース)

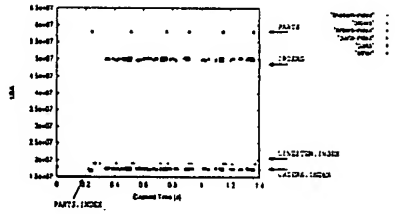


図 9: DBMS ポートのアクセス (PARTS ベース)

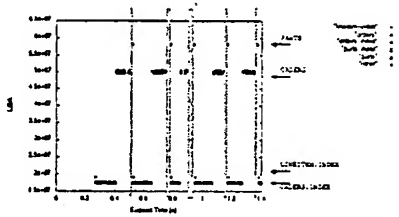


図 10: prefetch ポートのアクセス (PARTS ベース)

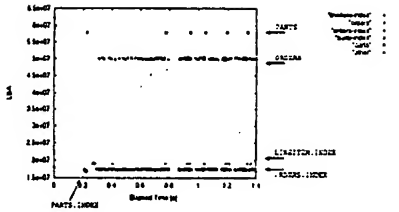


図 11: DBMS ポートのアクセス (PARTS.INDEX ベース)

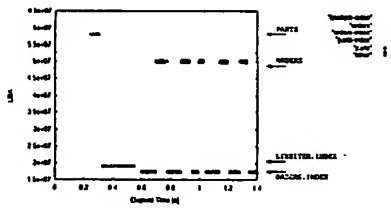


図 12: prefetch ポートのアクセス (PARTS.INDEX ベース)

前節で示した PARTS ベース、PARTS.INDEX ベースのプリフェッチのヒット状況、ミス状況を、それぞれ、図 13、図 14、図 15、図 16に示す。

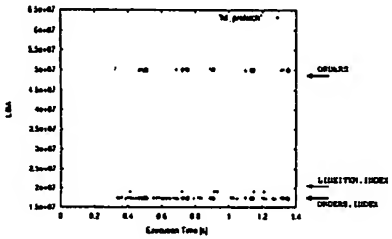


図 13: ヒット状況 (PARTS ベース)

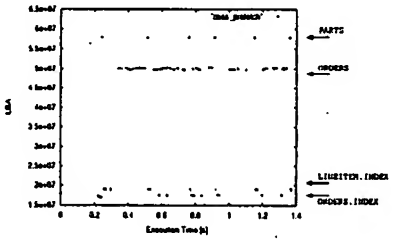


図 14: ミス状況 (PARTS ベース)

図 13、図 15において、プリフェッチがヒットすることにより I/O 時間が短縮され処理性能が向上している事がプロット点の集中により確認できる。図 16では、最初の約 0.6 秒間は、DBMS ポートからは、PARTS テーブルと LINEITEM.INDEX のプリフェッチが行われているあ、プリフェッチの効果が現れていないが、その後、DBMS ポートのプリフェッチが ORDERS.

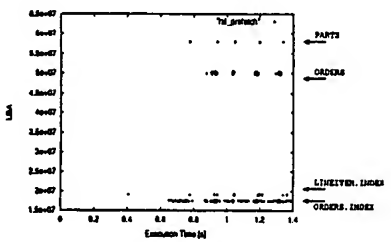


図 15: ヒット状況 (PARTS.INDEX ベース)

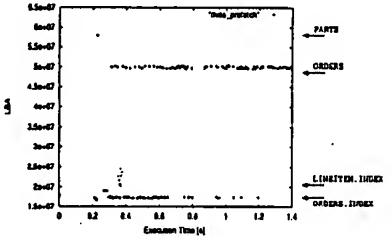


図 16: ミス状況 (PARTS.INDEX ベース)

INDEX と ORDERS に移る為、プリフェッチのヒット率が上昇し、処理時間が短縮されている。

表 4に、それぞれの先読みの深さに関するプリフェッチヒット率を示す。表から明らかな様に、プリフェッチの深さを深くする事により、ヒット率が向上していることが判る。

4.4 同時先読み IO 発行数の影響

一度にディスクがスケジューリングを行う事の出来る同時先読み IO 発行数が、処理時間にどの程度、影響を及ぼすのかを調べる為に、同時先読み IO 発行数に対する処理時間をプロットした。その図を図 17に示す。横軸が同時先読み IO 発行数、縦軸が実行時間で

	ヒット率 (%)
先読み無し	0
LINEITEM.INDEX ベース	77.4
PARTS ベース	82.8
PARTS.INDEX ベース	85.8

表 4: プリフェッチヒット率の比較 (同時先読み IO 発行数 30)

ある。

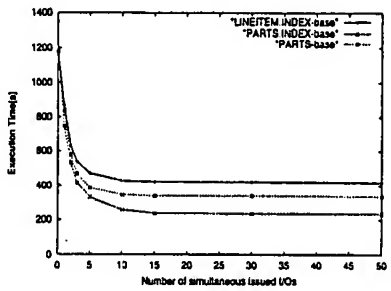


図 17: 同時先読み IO 発行数に対する実行時間

Q8 に於いて、同時先読み IO 発行数が 1 から 10 までの少ない範囲では、発行数が 1 つ増えるごとに処理時間の著しい削減がなされている。これは、ディスクアレイに於いて、ディスク 5 台にデータを分散して保持している為、それぞれのディスクに対し、同時に read コマンドが発行されることによって実現される並列処理による性能向上とスケジューリングの効果によるもの考えられる。しかし、同時先読み IO 発行数が 15 を超えた辺りからは、PARTS.INDEX ベースでは若干の性能向上が見られるが、その他の場合に於いては、処理性能は飽和し、同時先読み IO 発行数の増加に伴う処理時間の向上は見られない。

5 まとめと今後の課題

アプリケーションレベルの知識として、アクセスプランが与えられた場合、そのアクセスプランを用いたプリフェッチ機構を持つディスクに於いてどの程度の性能向上が得られるのかを示す為に、疑似的なプリフェッチシステムを構築し、評価実験を行った。その結果、アクセスプランによるプリフェッチを行うことにより、ヒット率が向上し、性能が大幅に改善される事が示された。今回は、ORACLE8 に関して実験を行ったが、その他の DBMS においても同じような結果が得られるのか、また、その他の問合せに於いても同様な効果が得られるのかなどの検討が必要と考えられる。今後は、これらの事を踏まえ、さらなる実装実験を行っていきたい。

謝辞

本研究に御協力賜わりました、日立製作所の大枝氏、松並氏に感謝致します。

参考文献

[1] D.Patterson et al. "Intelligent RAM(IRAM): the Industrial Setting, Applications, and Architectures". In *Proceedings of the International Conference on Computer Design*, 1997

[2] Erik Riedel, Garth Gibson, and Christors Faloutsos. "Active Storage For Large-Scale Data Mining and Multimedia". *Proceedings of the 24th VLDB Conference*, New York, USA, 1998

[3] Anurag Acharya, Mustafa Uysal, and Joel Saltz. "Active Disk: Programming Model, Algorithms and Evaluation". In *Proceedings of ASPLOS VIII*, page81-91,Oct 1998

[4] Kimberly Keeton, David A.Patterson, and Joseph M.Hellerstein. "A Case for Intelligent Disks(IDISKs)". *SIGMOD Record*,Volume27, Number3. August 1998

[5] D.A.Patterson, G.Gibson, and R.H.Katz. "A Case for Redundant Arrays of Inexpensive Disks(RAID)". In *Proceedings of ACMSIGMOD*, pp.109-116, Jun,1988

[6] A.Acharya, M.Uysal, and J.Saltz. "Active disks". Technical Report, TPC98-06, University of California, Santa Barbara, March 1998

[7] Mustafa Uysal, Anurag Acharya, and Joel Saltz. "An Evaluation of Architectural Alternatives for Rapidly Growing Datasets: Active Disks, Clusters, SMPs". TechnicalReport TRCS98-27, University of California, Santa Barbara, Oct 1998

[8] Cheetah Specification. *http://www.seagate.com/*

[9] Anurag Acharya, Mustafa Uysal, and Joel Saltz. "Structure and Performance of Decision Support Algorithms on Active Disks". Technical Report, TRCS98-28, University of California, Santa Barbara, Oct 1998

[10] Aaron Brown, David Oppenheimer, Kimberly Keeton, Randi Thomas, John Kubiatowicz, and David A.Patterson. "ISOTRE:Introspective Storage for Data-Intensive Network Services". *Proceedings of the 7th Workshop on Hot Topics in Operating Systems(HotOS-VII)* ,Rio Rico,Arizona, March 1999.

[11] *http://www.tpc.org/*

[12] 稲見聡, 喜連川優. "高機能ディスクにおけるアクセスプランを用いたプリフェッチ機構の一考察". *DEWS'99*